

Klasifikasi Soal Sejarah Tingkat SMA Berdasarkan Level Kognitif *Revised Bloom's Taxonomy* Menggunakan Algoritma *K-Nearest Neighbour Manhattan*

1st Rama Ariandi

Fakultas Rekayasa industri
Universitas Telkom
Bandung, Indonesia

ramaariandi@student.telkomuniversity.
ac.id

2nd Oktariani Nurul Pratiwi

Fakultas Rekayasa industri
Universitas Telkom
Bandung, Indonesia

onurulp@telkomuniversity.ac.id

3rd Riska Yanu Fa'rifah

Fakultas Rekayasa industri
Universitas Telkom
Bandung, Indonesia

riskayanu@telkomuniversity.ac.id

Abstrak—Pendidikan merupakan upaya terencana untuk mendorong siswa-siswi agar giat belajar guna mengembangkan bakat dan kecerdasannya melalui pembelajaran dengan melakukan berbagai latihan soal. Naskah soal biasanya berisi contoh soal dengan tingkat kesulitannya dan untuk dilakukan klasifikasi berdasarkan *Revised Bloom's Taxonomy* merupakan kasus yang tidak mudah jika dilakukan dengan cara manual. Peneliti ini menggunakan bantuan *machine learning* untuk mengklasifikasikan pertanyaan secara otomatis berdasarkan tingkat kognitif *Revised Bloom's Taxonomy* yang terfokus pada soal Sejarah Indonesia tingkat SMA dan level C1–C3 RBT. Dataset diperoleh dari berbagai latihan bank soal, ulangan harian, dan soal USBN yang terdapat di internet. Algoritma *K-Nearest Neighbor (KNN)* dengan pendekatan jarak *Manhattan* adalah metode yang digunakan pada penelitian ini. Selain itu, pembobotan kata sebagai bagian dari proses *TF-IDF* diterapkan terhadap dataset. Ketidakseimbangan data dari dataset yang diperoleh dalam penelitian ini diatasi dengan penggunaan metode *oversampling SMOTE*. Dataset tersebut kemudian diproses melalui *K-Fold Cross Validation* dengan nilai $k = 10$. Hasil akurasi penelitian yang telah dilakukan dengan algoritma *KNN* diperoleh nilai yang cukup baik dengan *precision 90%*, *recall 87%*, *F1-Score 87%* dan *accuracy 87%* dan teruji *SMOTE* dapat digunakan secara efektif dalam mengatasi *imbalance data*.

Kata kunci—klasifikasi soal, *KNN*, *manhattan*, *RBT*, *SMOTE*, *oversampling*

I. PENDAHULUAN

Isi Pendidikan pada hakikatnya merupakan suatu usaha terpolat untuk menciptakan suasana belajar dan proses pembelajaran agar pelajar secara aktif menghasilkan potensi diri untuk memiliki kecerdasan dan keterampilan yang dibutuhkan. Pendidikan juga adalah alur untuk mendapatkan pengetahuan dalam proses pembelajaran untuk mendapatkan pengetahuan yang luas dalam melatih pemahaman dan keterampilan diri dengan cara melakukan berbagai latihan soal [1]. Naskah soal merupakan sekumpulan soal yang telah terkelompok yang sesuai materi dan jenis dari ujian yang diberikan untuk pelajar berdasarkan soal – soal ujian yang telah dibuat sebelumnya [2].

Bloom's Taxonomy adalah metode klasifikasi yang dapat digunakan untuk memprediksi kemampuan pelajar dalam belajar sebagai penilaian dari kegiatan pembelajaran [3]. Mengapa menggunakan *Bloom's Taxonomy*? ini merupakan salah satu kerangka dasar ranah kognitif untuk klasifikasi yang telah digunakan setengah abad untuk penyusunan unsur pendidikan, penyusunan soal, dan kurikulum [4] dan *Bloom's Taxonomy* juga merupakan salah satu model taksonomi dengan tujuan sebagai acuan kurikulum pada aturan pendidikan di Indonesia .

Untuk melakukan klasifikasi soal berdasarkan *Bloom's Taxonomy* merupakan tugas yang sulit jika dilakukan secara manual [5], yang berdampak pada proses pengolahan data akan membutuhkan waktu yang cukup lama terhadap jumlah data kapasitas besar [6]. Dengan melakukan klasifikasi soal secara otomatis dengan bantuan *machine learning* berdasarkan *Bloom's Taxonomy*, penelitian ini bertujuan untuk mempermudah pengajar untuk menilai mengevaluasi hasil untuk mempertimbangkan tolak ukur kemampuan pelajar terhadap tingkat pengetahuan pelajar dari latihan soal yang diberikan [7].

Terdapat penelitian terdahulu yang menjadi rujukan pada penelitian lanjutan ini dalam melakukan klasifikasi soal berdasarkan level kognitif *Revised Bloom's Taxonomy* menggunakan algoritma *KNN* dengan pendekatan jarak *Manhattan*. Pada penelitian [2] yang menguji kombinasi algoritma *KNN* dan metode *feature selection chi-square* untuk mengklasifikasi soal berhasil mendapatkan tingkat akurasi yang cukup tinggi mencapai 79,36%. Selanjutnya pada penelitian [8] yang menguji kualitas butir soal berdasarkan *PISA* dan *Revised Bloom's Taxonomy*, hasil yang didapatkan sebagian besar masih belum memenuhi standar berdasarkan *PISA* dan *Revised Bloom's Taxonomy*. Penggunaan jarak *Manhattan* digunakan untuk penelitian ini dipercaya memiliki nilai akurasi yang lebih tinggi dibandingkan jarak *Euclidean* pada penelitian [9].

II. KAJIAN TEORI

A. Revised Bloom's Taxonomy

Awal tahun 1948, sekelompok pelajar melakukan klasifikasi dengan tujuan dan sasaran pendidikan. Tujuannya untuk mengembangkan sistem klasifikasi berdasarkan tiga domain yaitu kognitif, afektif, dan psikomotorik. Berjalannya waktu pada tahun 1950 pekerjaan domain kognitif selesai dan biasanya disebut kognitif Taksonomi Bloom [10] serta pada pendidikan, Taksonomi Bloom merupakan model yang digunakan untuk mengklasifikasikan teori sesuai dengan tingkat kognitifnya [11].

Keenam level kognitif *Revised Bloom's Taxonomy* di antaranya adalah *remembering* atau mengingat (C1), *understanding* atau memahami (C2), *applying* atau mengaplikasikan (C3), *analyzing* atau menganalisis (C4), *evaluation* atau mengevaluasi (C5) dan *creating* atau menciptakan (C6) [12]. Dari keenam tingkat RBT tersebut terbagi menjadi dua kelompok yaitu tingkat berpikir yang rendah meliputi aspek mengingat, memahami, dan mengaplikasikan dan tingkat berpikir yang tinggi meliputi aspek menganalisis, mengevaluasi, dan menciptakan [13].

B. K-Nearest Neighbors

K-Nearest-Neighbors merupakan metode klasifikasi non-parametrik, yang memiliki karakteristik sederhana dan efektif dalam menangani permasalahan pada pengklasifikasian [14]. Keunggulan metode ini dapat mencapai tingkat akurasi klasifikasi yang tinggi dalam menangani masalah yang memiliki distribusi tidak normal [15].

Penelitian ini menghitung jarak terdekat, dapat dengan menggunakan metode jarak *Manhattan* yang didefinisikan sebagai berikut [16]:

$$d(X, Y) = \sum_{i=1}^n |X_i - Y_i| \quad (1)$$

Keterangan:

d : distance / jarak
n : jumlah fitur pada vektor data
X : data sampel
Y : data tes

C. Klasifikasi Pertanyaan

Klasifikasi pertanyaan merupakan metode digunakan untuk menganalisis pertanyaan serta pelabelan pertanyaan berdasarkan jenis jawaban yang diharapkan. Tujuan dari metode klasifikasi pertanyaan adalah dapat melakukan pemetaan label pertanyaan berdasarkan jenis jawaban yang diharapkan. Metode ini dapat mengatasi masalah serta meningkatkan kinerja pada *machine learning* [17].

D. Text preprocessing

Text preprocessing adalah proses awal untuk melakukan pemrosesan data berupa teks yang tidak terstruktur menjadi terstruktur. Dimana text preprocessing dapat digunakan untuk penerapan aplikasi dalam melakukan pemilihan data berbentuk teks dari setiap [18], serta proses ini dapat digunakan untuk pengklasifikasian. Terdapat empat tahapan yang terdiri:

1. *Case Folding*,
2. *Tokenization*,
3. *Filtering*,
4. *Stemming*.

E. Text Mining

Text Mining atau penambangan teks merupakan proses penerapan algoritma dan metode machine learning yang memiliki tujuan untuk menemukan pola atau struktur dari sebuah data. Pada dasarnya proses ini dilakukan dengan mengolah teks terlebih dahulu untuk menemukan pola tertentu [19].

F. Confusion matrix

Confusion matrix merupakan alat ukur model evaluasi performansi yang digunakan untuk mengukur performa suatu algoritma. Tujuan model ini adalah membandingkan hasil klasifikasi yang dilakukan dengan data-data berdasarkan tabel *confusion matrix* [20]. Berikut merupakan tabel *actual values* dan *prediction values* dari model *confusion matrix* [21].

TABEL 1
HAHAHAH

		Actual Values	
		True	False
Prediction Values	True	TP (True Positive)	FP (False Positive)
	False	FN (False Negative)	TN (True Negative)

G. Synthetic Minority Oversampling Technique (SMOTE)

SMOTE merupakan metode *oversampling* untuk menyeimbangkan jumlah data pada kelas minoritas atau kelas yang memiliki jumlah paling sedikit dengan meningkatkan jumlah data secara acak [22] sehingga jumlah data kelas minoritas menjadi seimbang terhadap jumlah pada kelas mayoritas atau kelas yang memiliki jumlah paling banyak [23].

H. Term Frequency Inverse Document Frequency (TF-IDF)

TF-IDF merupakan metode proses pembobotan yang dilakukan untuk memperhitungkan bobot terhadap setiap kata (term) yang dihasilkan pada tahap preprocessing [24]. Keunggulan metode pengukuran statistik ini dapat digunakan dalam melakukan evaluasi sebuah kata itu penting atau tidaknya di dalam sebuah dokumen [25]. Berikut adalah rumus untuk menghitung TF-IDF [26]:

- a) TF (*Term Frequency*)

$$tf = 1 + \log_{10} f_t \quad (2)$$

Keterangan:

tf = term frekuensi

- b) IDF (*Inverse Document Frequency*)

$$idf = \log_{10} \frac{D}{df} \quad (3)$$

Keterangan:

idf = inverse dokumen frekuensi

D = total dokumen

Df = frekuensi dokumen dari term

c) Menghitung bobot dokumen (W)

$$W = tf \times idf \quad (4)$$

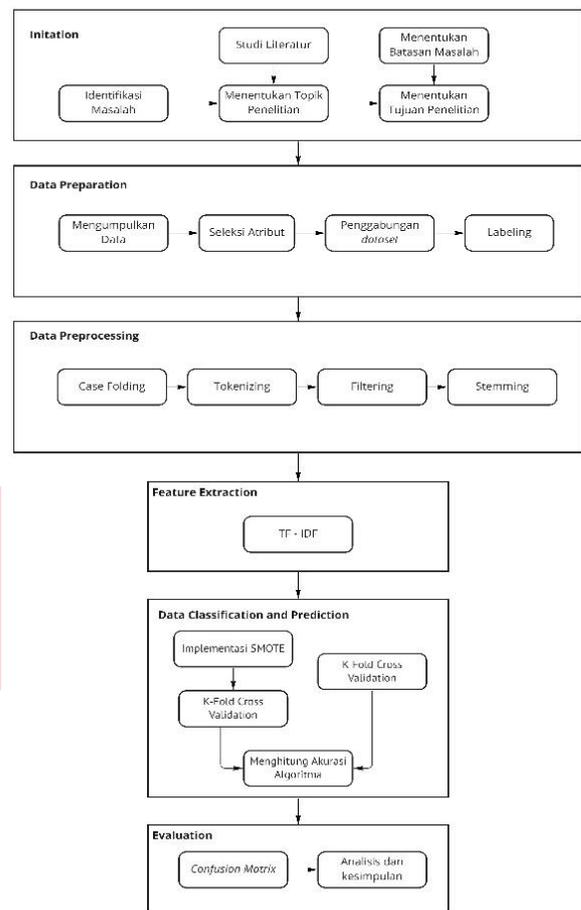
I. Cross validation

Cross validation merupakan metode model validasi yang digunakan untuk menilai seberapa jauh hasil analisis statistik dalam menggeneralisasi suatu data independen dengan jumlah yang besar [27]. Metode ini digunakan untuk menentukan data training serta data testing pada proses klasifikasi [28].

K-Fold Cross Validation adalah metode pendekatan validasi untuk menentukan parameter terbaik dari sebuah model klasifikasi [29]. Pada metode validasi ini, nilai k yang sering digunakan adalah *10-fold* ($k=10$). Manfaat dari metode ini adalah setiap record terjadi hanya sekali dalam set pengujian data [30].

III. METODE

Pada penelitian ini memiliki beberapa tahapan yaitu inialisasi, pengolahan data, ekstrasi hasil, klasifikasi dan kesimpulan. Dapat digambarkan sistematis penyelesaian masalah pada gambar di bawah ini.



GAMBAR 1
SISTEMATIKA PENULISAN

A. Initiation

Tahap *initiation* meliputi beberapa proses diantaranya identifikasi masalah dan studi literatur untuk penentu topik dari penelitian. Selanjutnya menentukan batasan masalah sebagai penentu tujuan dari penelitian ini.

B. Data Preparation

Tahap berikutnya adalah *data preparation*. Tahap ini meliputi proses mengumpulkan data, seleksi atribut, menggabungkan *dataset* dan *labelling* data. Penelitian ini memperoleh data berupa kumpulan soal bertipe pilihan ganda Sejarah Indonesia tingkat SMA seperti soal USBN, ulangan harian, dan bank latihan soal yang terdapat di internet. Semua data yang telah didapatkan dan dikumpulkan, lalu data di ekstrak ke format *csv* dan menjadi *dataset*. Dari data soal pertanyaan yang telah dikumpulkan pada *dataset* terfokus pada C1-C3 yang berjumlah 246 butir soal. Selanjutnya proses penghilangan pilihan ganda pada setiap soal, yang memiliki tujuan untuk menghilangkan atribut – atribut yang dapat mempengaruhi duplikasi data dan data di *export* dengan format *xlsx*. Peneliti melakukan *labelling data* dengan memberikan label berdasarkan level kognitif RBT secara manual.

C. Data Preprocessing

Pada tahap *data preprocessing* dimana data akan melalui empat proses yaitu *case folding*, *tokenization*, *filtering*, dan *stemming* untuk menjadi data yang dapat diolah pada tahap berikutnya yaitu *feature extraction* yang menggunakan *tools Python* dengan bantuan *library* sastrawi dan NLTK.

D. Feature Extraction

Pada tahap *feature extraction*, penggunaan metode TF-IDF dilakukan dengan tujuan untuk pembobotan setiap kata karena algoritma tidak dapat menangani data berbentuk teks secara langsung, oleh karena itu perlu diubah ke bentuk numerik. Tahapan ini dilakukan pembobotan terhadap jumlah kemunculan term pada dokumen (*Term Frequency*) dan invers dari frekuensi dokumen (*Inverse Document Frequency*) akan dihitung. Pembobotan pada setiap kata pada TF-IDF merupakan hasil dari proses *stemming* dan *output* tahapan ini adalah setiap kata dari soal-soal akan menjadi data berbentuk numerik yang dibutuhkan untuk tahap selanjutnya.

E. Data Classification and Prediction

Tahap selanjutnya adalah *data classification and prediction* dengan beberapa implementasi diantaranya metode *oversampling* SMOTE, *K-Fold Cross Validation*, dan perhitungan algoritma klasifikasi menggunakan KNN dengan pendekatan jarak *Manhattan*.

F. Evaluation

Pada tahap terakhir yaitu *evaluation*, dimana tahapan ini mengimplementasikan *confusion matrix* yang dibutuhkan kan untuk mengukur performa algoritma KNN terhadap dataset yang telah selesai diproses dilakukan dengan maupun tanpa SMOTE dan menganalisis hasil yang diperoleh untuk kesimpulan dari penelitian ini,

IV. HASIL DAN PEMBAHASAN

Data yang diperoleh berupa kumpulan soal bertipe pilihan ganda berbentuk teks yang tidak ada rumus dalam setiap soal Sejarah Indonesia tingkat SMA seperti soal USBN, ulangan harian, dan bank latihan soal. Keseluruhan soal dari 304 soal berdasarkan RBT C1-C6 dan terfokus pada RBT C1-C3 yang memiliki jumlah 246 soal saja. Peneliti melakukan penggabungan soal serta menghilangkan pilihan ganda pada setiap soal yang menjadi pertanyaan saja dengan tujuan untuk menghilangkan atribut – atribut yang dapat mempengaruhi duplikasi data. Selanjutnya ketika data telah dilakukan penghilangan pilihan ganda pada setiap soal, selanjutnya data di *export* dengan format *xlsx* untuk proses selanjutnya yaitu proses *labeling*.

A. Labeling

Pada tahap ini, peneliti melakukan labeling dataset dengan memberikan label berdasarkan tingkat kognitif RBT secara manual seperti contoh pada tabel di bawah.

TABEL 1
CONTOH SOAL BESERTA LABELNYA

Soal	Label
Hal yang menjadi faktor internal pendorong munculnya nasionalisme Indonesia adalah...	<i>Remembering/ C1</i>
Pasukan Jepang memiliki semangat juang dan disiplin tinggi, sebab pasukan Jepang berpegang teguh pada kode etik keprajuritan yang disebut...	<i>Understanding/ C2</i>
Berikut kelompok-kelompok para pejuang yang akan menjelang kemerdekaan, kecuali...	<i>Applying/ C3</i>

Untuk menentukan C1-C3 dalam proses labeling, penulis menganalisa kembali kata kunci untuk pedoman dalam

melakukan labeling soal Sejarah Indonesia berdasarkan RBT [13]. Berikut kata kunci yang digunakan untuk melakukan labeling.

TABEL 2.
KATA KUNCI

Label	Kata Kunci
C1	apa, kapan, di mana, yang, siapa, mengapa. yakni, disebut, adalah, tujuan, sebagai, oleh + siapa
C2	kecuali, untuk, sebab-dampak, dalam rangka, yaitu, karena, alasan, sebab + oleh, berupa, contoh, antara + antar, dengan, jika
C3	salah-satu, antara lain, seperti, rangka, faktor, berikut ini merupakan – kecuali

B. Data Preprocessing

Pada tahap *data preprocessing*, dimana data akan melalui empat proses yaitu *case folding*, *tokenization*, *filtering*, dan *stemming* untuk menjadi data yang dapat diolah pada tahap berikutnya yaitu TF-IDF.

TABEL 3
HASIL PREPROCESSING

<i>Raw Data</i>	Sumber Kerajaan Tarumanegara yang lokasi penemuannya terletak di wilayah Provinsi Banten adalah
<i>Case Folding</i>	kebudayaan hindu merupakan perpaduan antara kebudayaan arya dan kebudayaan
<i>Tokenizing</i>	['sumber', 'kerajaan', 'tarumanegara', 'yang', 'lokasi', 'penemuannya', 'terletak', 'di', 'wilayah', 'provinsi', 'banten', 'adalah']
<i>Filtering/Stopword Removal</i>	['sumber', 'kerajaan', 'tarumanegara', 'lokasi', 'penemuannya', 'terletak', 'wilayah', 'provinsi', 'banten']
<i>Stemming</i>	['sumber', 'raja', 'tarumanegara', 'lokasi', 'temu', 'letak', 'wilayah', 'provinsi', 'banten']

C. Pembobotan TF-IDF

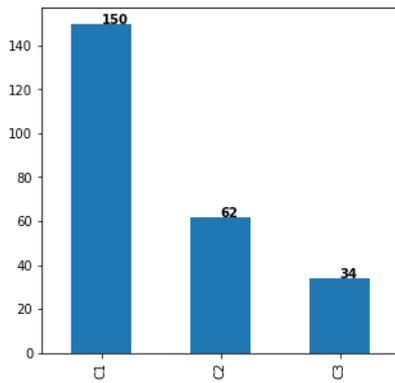
Pada tahapan TF-IDF, data berupa soal pertanyaan diproses dengan dilakukan pembobotan pada setiap kata setelah melalui proses tahapan *data preprocessing*. Berikut contoh hasil dari tahapan proses TF-IDF yang dapat dilihat pada tabel di bawah ini.

TABEL 4
CONTOH HASIL TF-IDF

<i>Stemming</i>	TF-IDF
['sumber', 'raja', 'tarumanegara', 'lokasi', 'temu', 'letak', 'wilayah', 'provinsi', 'banten']	'sumber': 0.1713977862096481 'raja': 0.4576707972013858 'tarumanegara': 0.5346871505969352 'lokasi': 0.5346871505969352 'temu': 0.41261911852270083 'letak': 0.4328770692775847 'wilayah': 0.3806544438058363 'provinsi': 0.5346871505969352 'banten': 0.3675674398440159

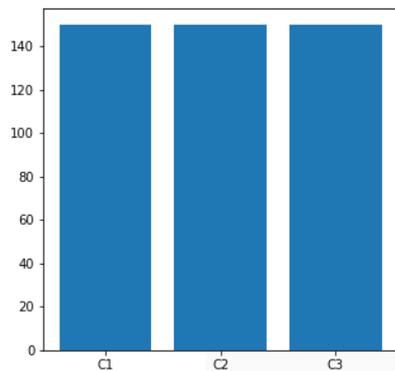
D. Implementasi SMOTE

Implementasi SMOTE digunakan untuk menangani *imbalance data* pada *dataset* soal. Untuk mengetahui *dataset* mengalami *imbalance* adalah terdapat perbandingan rasio yang jauh antara label mayoritas dan minoritas. Berikut merupakan visualisasi data sebelum SMOTE



GAMBAR 2
HASIL DATA SEBELUM SMOTE

Pada gambar di atas dapat dilihat bahwa data mengalami ketidakseimbangan antar label satu dengan lainnya. Label C1 memiliki jumlah yang paling banyak sebesar 150 butir soal, label C2 memiliki jumlah yaitu 62 butir soal dan C3 memiliki jumlah yang paling sedikit yaitu 34 butir soal. Oleh karena itu metode *oversampling* SMOTE digunakan untuk mengatasi *imbalance* data dengan menyeimbangkan jumlah data kelas minoritas supaya kelas minoritas memiliki jumlah yang seimbang dengan kelas mayoritas [23]. Berikut visualisasi data yang telah diimplementasikan dengan SMOTE.



GAMBAR 3
HASIL DATA SETELAH SMOTE

E. Implementasi *K-Fold Cross Validation* dan KNN

Pada implementasi *K-Fold Cross Validation*, hasil kedua dataset dengan maupun tanpa menggunakan SMOTE digunakan K-Fold pada penelitian ini dengan nilai $k = 10$. Nilai K-Fold dengan jumlah $k = 10$ digunakan secara umum untuk mendapatkan estimasi hasil akurasi [31].

Selanjutnya tahapan ini dilakukan implementasi menggunakan algoritma KNN. Penelitian ini terfokus pada algoritma KNN dengan pendekatan jarak *Manhattan*. KNN dengan pendekatan jarak *Manhattan* dapat meningkatkan nilai akurasi yang tinggi pada proses klasifikasi [32]. Berikut hasil perbandingan *K-Fold* sebelum dan setelah SMOTE dengan mengimplementasikan *K-Fold Cross Validation* dan algoritma KNN.

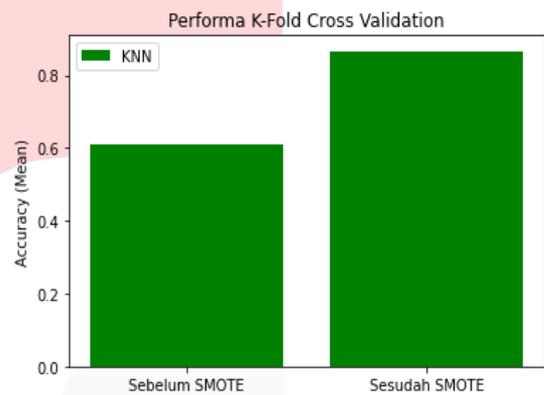
TABEL 5
HASIL *K-FOLD* KNN SEBELUM SMOTE

Fold ke -	1	2	3	4	5	6	7	8	9	10
Akurasi %	56 %	72 %	72 %	68 %	56 %	68 %	37 %	50 %	62 %	67 %

TABEL 6
HASIL *K-FOLD* KNN SETELAH SMOTE

Fold ke -	1	2	3	4	5	6	7	8	9	10
Akurasi %	91 %	89 %	84 %	93 %	91 %	84 %	84 %	80 %	84 %	84 %

Hasil rata-rata akurasi dari implementasi menggunakan SMOTE didapatkan dengan jumlah 87% sedangkan hasil implementasi tanpa menggunakan SMOTE adalah 61%. Berikut visualisasi data yang dapat dilihat pada Gambar V.3 dan dari hasil tersebut teruji bahwa implementasi *K-Fold Cross Validation* dengan SMOTE dapat meningkatkan akurasi algoritma klasifikasi yang cukup tinggi pada penelitian ini.



GAMBAR 4
PERFORMA *K-FOLD* CROSS VALIDATION

F. *Confusion Matrix*

Pada tahap terakhir yaitu evaluasi performansi. Evaluasi performansi digunakan untuk mengukur performa dari algoritma KNN terhadap dataset yang telah dilakukan dengan maupun tanpa SMOTE. Berikut tabel merupakan hasil *confusion matrix* tanpa SMOTE.

TABEL 7
HASIL *CONFUSION MATRIX* TANPA SMOTE

Predicted/ True	C1	C2	C3	All
C1	150	0	0	150
C2	62	0	0	62
C3	34	0	0	34
All	246	0	0	246

Pada tabel di atas bahwa hasil dari *confusion matrix* tanpa SMOTE dapat diketahui model memprediksi true positive (TP) pada label C1 dengan jumlah 150. Algoritma KNN tidak memprediksi label C2 dan C3 sehingga hasil yang didapatkan 0. Semua data berlabel C2 dan C3 diprediksi ke kelas mayoritas yaitu C1. Berikutnya adalah tabel merupakan hasil *confusion matrix* dengan SMOTE.

TABEL 8
HASIL *CONFUSION MATRIX* DENGAN SMOTE

Predicted/ True	C1	C2	C3	All
C1	146	2	2	150
C2	46	104	0	150

C3	10	0	140	150
All	202	106	142	450

Berdasarkan tabel di atas, bahwa model memprediksi true positive (TP) label C1 berjumlah 146, label C2 dengan total 104 dan label C3 berjumlah 140. Kesimpulan dari tabel di atas diketahui model memprediksi lebih banyak pada label C1. Setelah mendapatkan hasil *confusion matrix*, selanjutnya tabel merupakan hasil evaluasi performansi yang diperoleh.

TABEL 9
HASIL EVALUASI PERFORMANSI

	Sebelum SMOTE	Setelah SMOTE
Precision	37%	90%
Recall	61%	87%
F1-Score	46%	87%
Accuracy	61%	87%

Pada tabel di atas, nilai *precision*, *recall*, dan *f1-score* sebelum dan setelah SMOTE berdasarkan nilai *weighted average* memiliki perbandingan nilai yang signifikan jauh dan hasil setelah SMOTE dapat dilihat bahwa mampu meningkatkan tingkat akurasi dari performa model klasifikasi. Hasil *precision* merupakan hasil positif terhadap jumlah prediksi target yang memiliki nilai positif. Nilai *precision* yang didapatkan sebelum dan setelah SMOTE adalah 37% dan 90%. Nilai *recall* merupakan hasil target positif terhadap jumlah data positif, dan hasil yang diperoleh sebelum dan setelah SMOTE adalah 61% dan 87%. Selanjutnya nilai *f1-score* merupakan hasil alternatif akurasi dan hasil yang diperoleh sebelum SMOTE adalah 46% dan setelah SMOTE 87%. Selanjutnya, nilai *accuracy* merupakan hasil akurasi model klasifikasi dan hasil yang diperoleh sebelum SMOTE adalah 61% dan setelah SMOTE adalah 87%.

Berdasarkan penelitian yang telah dilakukan bahwa dengan menggunakan algoritma KNN dengan pendekatan jarak *Manhattan* dapat meningkatkan nilai akurasi yang cukup tinggi setelah implementasi menggunakan SMOTE. Penggunaan SMOTE pada penelitian [33] mampu meningkatkan kinerja dari algoritma klasifikasi jika dibandingkan sebelum menggunakan SMOTE. Metode *oversampling* ini beroperasi menyeimbangkan jumlah data kelas minoritas dengan meningkatkan jumlah data secara acak sehingga jumlah data pada kelas minoritas seimbang dengan data kelas mayoritas [22].

V. KESIMPULAN

Dari penelitian yang telah dilakukan, *dataset* soal Sejarah Indonesia tingkat SMA yang digunakan untuk proses klasifikasi secara otomatis menggunakan metode *machine learning* algoritma KNN pendekatan jarak *Manhattan* dengan bantuan *tools* Python 3.9 diperoleh dari internet yang telah dikumpulkan serta di import ke *google drive* untuk dapat diakses dengan mudah. Jumlah diperoleh adalah 246 butir soal yang terfokus pada tingkat RBT C1-C3 dan dilakukan labeling secara manual berdasarkan level kognitif RBT. Dalam mengatasi *imbalance data* terhadap *dataset* soal Sejarah Indonesia C1-C3 yang digunakan, peneliti menggunakan metode *oversampling* SMOTE untuk mengatasinya. Dari hasil penelitian yang didapatkan bahwa nilai akurasi algoritma KNN dengan *K-Fold* nilai $k = 10$ setelah SMOTE adalah 87%, sedangkan *K-Fold* sebelum SMOTE memperoleh nilai

akurasi yaitu 61%. Sehingga penggunaan setelah SMOTE pada penelitian ini mampu mengoptimalkan kinerja proses klasifikasi yang bermakna dapat meningkatkan nilai akurasi sebesar 26%. Selanjutnya, peneliti menyarankan agar dapat dilakukan penelitian lanjutan menggunakan algoritma klasifikasi berbagai macam dan metode *oversampling* lainnya untuk memperoleh tingkat akurasi algoritma yang paling baik.

REFERENSI

- [1] Adib Fathoni, "Wawasan Pendidikan," *J. Ushuluddin*, vol. 2, p. 17, 2019.
- [2] I. Listiowarni and N. Puspa Dewi, "Pemanfaatan Klasifikasi Soal Biologi Cognitive Domain Bloom's Taxonomy Menggunakan KNN Chi-Square Sebagai Penyusunan Naskah Soal," *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 11, no. 2, pp. 186–197, 2020, doi: 10.31849/digitalzone.v11i2.4798.
- [3] E. Sujoko and I. P. A. Darmawan, "Revisi Taksonomi Pembelajaran Benyamin S. Bloom," vol. 29, no. 1, pp. 30–39, 2013.
- [4] I. Gunawan and A. R. Paluti, "Taksonomi Bloom – Revisi Ranah Kognitif," *E-Journal.Unipma*, vol. 7, no. 1, pp. 1–8, 2017, [Online]. Available: <http://e-journal.unipma.ac.id/index.php/PE>.
- [5] S. F. Kusuma, A. Heriadi, and M. F. Naufal, "Menggunakan Algoritma Svm," pp. 365–370, 2017.
- [6] A. Aninditya, M. A. Hasibuan, and E. Sutoyo, "Text mining approach using TF-IDF and naive bayes for classification of exam questions based on cognitive level of bloom's taxonomy," *Proc. - 2019 IEEE Int. Conf. Internet Things Intell. Syst. IoTaIS 2019*, no. c, pp. 112–117, 2019, doi: 10.1109/IoTaIS47347.2019.8980428.
- [7] N. E. Adams, "Bloom's taxonomy of cognitive learning objectives," *J. Med. Libr. Assoc.*, vol. 103, no. July, pp. 152–153, 2015, [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4511057/>.
- [8] Y. Pantiwati and F. H. Permana, "Analisis butir soal oleh mahasiswa S1 pendidikan biologi universitas muhammadiyah malang (UMM) berdasarkan PISA dan taksonomi bloom revisi," *Publ. Ilm. UMS*, pp. 707–716, 2017, [Online]. Available: <https://publikasiilmiah.ums.ac.id/xmlui/handle/11617/9550>.
- [9] R. K. Dinata, H. Akbar, and N. Hasdyna, "Algoritma K-Nearest Neighbor dengan Euclidean Distance dan Manhattan Distance untuk Klasifikasi Transportasi Bus," *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 104–111, 2020, doi: 10.33096/ilkom.v12i2.539.104-111.
- [10] W. Huit, "Bloom et al.'s taxonomy of the cognitive domain," *Educ. Psychol. Interact.*, 2011.
- [11] T. J. Lasley, "Bloom's Taxonomy," *Encyclopedia of Educational Reform and Dissent*. 2013, doi: 10.4135/9781412957403.n51.
- [12] S. Fatmawati and M. Pd, "Perumusan Tujuan Pembelajaran Dan Soal Kognitif Berorientasi Pada Revisi Taksonomi Bloom Dalam Pembelajaran Fisika," *Edu Sains J. Pendidik. Sains dan Mat.*, vol. 1, no. 2, 2013, Accessed: Jan. 06, 2022. [Online].

- Available: <https://e-journal.iainpalangkaraya.ac.id/index.php/edusains/article/view/13>.
- [13] Minarto, "Pemunculan Tingkat Kesulitan Soal Pada Tes Penjurusan Menggunakan Revides Bloom Taxonomi (RBT) Di SMAN 1 Bangorejo Dengan Aplikasi WINGEN3," *J. INCARE*, vol. 01, no. 01, pp. 17–27, 2020.
- [14] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN model-based approach in classification," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2888, pp. 986–996, 2003, doi: 10.1007/978-3-540-39964-3_62.
- [15] S. Baek and K.-M. Sung, "Fast K-nearest-neighbour search algorithm for nonparametric classification," vol. 36, no. 21, 2000, doi: 10.1049/el:20001249.
- [16] M. E. Saputra, H. Mawengkang, and E. B. Nababan, "Gini Index with Local Mean Based for Determining K Value in K-Nearest Neighbor Classification," *J. Phys. Conf. Ser.*, vol. 1235, no. 1, 2019, doi: 10.1088/1742-6596/1235/1/012006.
- [17] D. Metzler and W. B. Croft, "Analysis of statistical question classification for fact-based questions," *Inf. Retr. Boston.*, vol. 8, no. 3, pp. 481–504, 2005, doi: 10.1007/s10791-005-6995-3.
- [18] L. Hermawan and M. Bellaniar Ismiati, "Pembelajaran Text Preprocessing berbasis Simulator Untuk Mata Kuliah Information Retrieval," *J. Transform.*, vol. 17, no. 2, p. 188, 2020, doi: 10.26623/transformatika.v17i2.1705.
- [19] A. Hotho, A. Nürnberger, and G. Paass, "A Brief Survey of Text Mining," *undefined*, 2005.
- [20] Adi Nugroho, Agustinus Bimo Gumelar, Adri Gabriel Sooai, Dyana Sarvasti, and Paul L Tahalele, "Perbandingan Performansi Kinerja Algoritma Pengklasifikasian Terpandu Untuk Kasus Penyakit Kardiovaskular," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 5, pp. 998–1006, Oct. 2020, doi: 10.29207/RESTI.V4I5.2316.
- [21] I. M. A. Agastya, "Pengaruh Stemmer Bahasa Indonesia Terhadap Peforma Analisis Sentimen Terjemahan Ulasan Film," *J. Tekno Kompak*, vol. 12, no. 1, p. 18, 2018, doi: 10.33365/jtk.v12i1.70.
- [22] R. Azmatul Barro, I. D. Sulvianti, and M. Afendi, "Penerapan Synthetic Minority Oversampling Technique (Smote) Terhadap Data Tidak Seimbang Pada Pembuatan Model Komposisi Jamu," *Xplore J. Stat.*, vol. 1, no. 1, pp. 1–6, 2013.
- [23] A. N. Kasanah, M. Muladi, and U. Pujiyanto, "Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 2, pp. 196–201, 2019, doi: 10.29207/resti.v3i2.945.
- [24] D. A. Prabowo, M. Fhadli, M. A. Najib, H. A. Fauzi, and I. Cholissodin, "TF-IDF-Enhanced Genetic Algorithm Untuk Extractive Automatic Text Summarization," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 3, p. 208, 2016, doi: 10.25126/jtiik.201633217.
- [25] J. A. Septian, T. M. Fachrudin, and A. Nugroho, "Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor," *J. Intell. Syst. Comput.*, vol. 1, no. 1, pp. 43–49, Aug. 2019, doi: 10.52985/INSYST.V1I1.36.
- [26] R. R. A. Siregar, F. A. Sinaga, and R. Arianto, "Aplikasi Penentuan Dosen Penguji Skripsi Menggunakan Metode TF-IDF dan Vector Space Model," *Comput. J. Comput. Sci. Inf. Syst.*, vol. 1, no. 2, p. 171, 2017, doi: 10.24912/computatio.v1i2.1014.
- [27] F. Tempola, M. Muhammad, and A. Khairan, "Perbandingan Klasifikasi Antara KNN dan Naive Bayes pada Penentuan Status Gunung Berapi dengan K-Fold Cross Validation," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 5, p. 577, 2018, doi: 10.25126/jtiik.201855983.
- [28] R. E. Putri, Suparti, and R. Rahmawati, "Perbandingan Metode Klasifikasi Naive Bayes Dan K-Nearest Neighbour Pada Analisis Data Status Kerja di Kab.Demak," *J. Gaussian*, vol. 3, pp. 831–838, 2014.
- [29] Okfalisa, I. Gazalba, Mustakim, and N. G. I. Reza, "Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification," *Proc. - 2017 2nd Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng. ICITISEE 2017*, vol. 2018-Janua, pp. 294–298, 2018, doi: 10.1109/ICITISEE.2017.8285514.
- [30] A. Muhidin and A. Burhan, "Klasifikasi Tingkat Produksi Tanaman Padi Kabupaten Karawang Menggunakan Metode Naive Bayes dan K-Fold Cross Validation," *J. Teknol. Pelita Bangsa- SIGMA*, vol. 8, no. 2, pp. 2407–3903, 2018.
- [31] A. Rohman and M. Rochcham, "Komparasi Metode Klasifikasi Data Mining Untuk Prediksi Kelulusan Mahasiswa," *Neo Tek.*, vol. 5, no. 1, pp. 23–29, 2019, doi: 10.37760/neoteknika.v5i1.1379.
- [32] K. Latifah, "Kombinasi-Algoritma-K-Nn-Dan-Manhattan," *J. Inform.*, vol. 1, no. 1, pp. 49–58, 2015.
- [33] S. Susanti, "Klasifikasi Kemampuan Perawatan Diri Anak dengan Disabilitas Menggunakan SMOTE Berbasis Neural Network," *J. Inform.*, vol. 6, no. 2, pp. 175–184, 2019, doi: 10.31311/ji.v6i2.5798.